Available online at www.sciencedirect.com
SciVerse ScienceDirect
ELSEVIER
Procedia CIRP 7 (2013) 222 – 227
www.elsevier.com/locate/procedia

Forty Sixth CIRP Conference on Manufacturing Systems 2013

Planning of remote laser welding processes

Gábor Erdős[a]*, Zsolt Kemény[a], András Kovács[a], József Váncza[a,b]

[a]*Fraunhofer Project Center on Production Management and Informatics, Computer and Automation Research Institute, Hungarian Academy of Sciences, H-1111 Budapest, Kende u. 13-17, Hungary*
[b]*Department of Manufacturing Science and Technology, Budapest University of Technology and Economics, H-1111 Budapest, Egry József u. 1, Hungary*

* Corresponding author. Tel.: +36-1-2796299; fax: +36-1-4667503. *E-mail address*: {erdos,zsolt.kemeny,andras.kovacs,vancza}@sztaki.mta.hu

Abstract

The paper discusses the technical background of the remote laser welding (RLW) technology, its novel opportunities and implications for planning processes. Our ultimate goal is to develop a complete off-line programming toolbox for RLW which can provide an automated method for computing close-to-optimal robot programs. We suggest a workflow for the complete planning process, and propose new models and algorithms for solving the sequencing of welding tasks in conjunction with path planning, as well as for generating the inverse kinematics of the robot. The paper summarizes results of first computational experiments in an automotive case study using an industrial robot. The proposed method leads to a substantial reduction in the cycle time of the welding operation compared to an earlier approach.

Selection and peer-review under responsibility of Professor Pedro Filipe do Carmo Cunha

Keywords: Planning; robot; task sequencing; remote laser welding

1. Introduction

One of the most significant current technological trends in car body making is the spreading application of *remote laser welding* (RLW) technology. RLW operations are performed from a distant point, by means of a laser beam that is emitted from a scanner mounted on the arm of an industrial *robot*. In contrast to traditional *resistance spot welding* (RSW), this contactless technology has to comply with much less accessibility constraints and can, at the same time, operate at higher speed. However, the new technology is much more expensive. Hence, replacing RSW with RLW technology is feasible only if the *cycle time* of the products can be considerably decreased [4][10].

The paper presents this relatively new technology, together with its novel opportunities and implications for *planning* processes. Our ultimate goal is to develop an *off-line programming* toolbox for RLW with semi-automated methods that are appropriate for computing close-to-optimal robot programs [16]. The methods should be applicable under industrial conditions.

After discussing the technological background and reviewing related literature, Section 2 presents our particular assumptions and problem formulation. The overall workflow of planning is discussed in Section 3. Here, a new method is also suggested for solving the crux of the problem which is the sequencing of welding tasks in conjunction with path planning. Further on, we explain how the result of path planning is transformed by inverse kinematics into a robot program. Section 4 gives a short account of our first computational experiments. According to these results the proposed method leads to a substantial reduction in the cycle time of the welding operation compared to an earlier approach.

1.1. Technical background

Laser welding can be regarded as a special way of applying the heat required for melting the materials meant to be joined—this can be done for a continuous seam, as well as for a single spot. Using a laser beam to this end has a number of advantages over more conventional forms of welding like RSW. First, laser welding eliminates the need of direct tool contact with the workpiece that implies large obstacle-free tool and robot sweep volumes for conventional technologies. Laser welding requires only a narrow straight line of sight free

2212-8271
Selection and peer-review under responsibility of Professor Pedro Filipe do Carmo Cunha
doi:10.1016/j.procir.2013.05.038

of obstruction, allowing welding in tight corners many conventional tools would not reach. Second, RLW is performed from a distance, usually with a scanner that uses mirrors and lenses to set the orientation and focal length of the beam. These components can act much faster (i.e., have a larger control bandwidth due to their small inertia) which can speed up the entire process (both tracing the welding locations, as well as repositioning between seams). Also, welding can take place with all robot joints and scanner elements continuously moving, resulting in smaller control transients, implying energy savings and allowing faster operation.

Nevertheless, laser welding does have its specific application constraints, resulting from the nature of the welding beam and the properties of the scanner head:

- Full visibility has to be ensured for the entire length of the seam. This has to be taken in consideration for the planning of fixtures, robot motion (avoiding occluding segments), and layout of potential visibility obstacles (even parts of the workpiece).
- Due to surface penetration properties, the beam-to-surface inclination angle has to remain within technologically prescribed bounds.
- Scanner heads may be very limited in beam deflection angle and focal length—this has to be observed when the rest of the robot motion is calculated.

The costs of an RLW cell are one more application constraint of a different kind. Switching to this new technology is only justified if expected advantages like reduction of cycle time or quality improvement balance out the costs in the given manufacturing context.

1.2. Related works

Laser welding is a complex process typically performed by a welding robot, and often relying on extensive sensory feedback. Many of its underlying planning and execution control aspects have received attention in the research community in recent years.

A number of publications (e.g., [17][18]) deal with the *process* of exposing the workpiece to the laser beam, giving insight into constraints of the physical process (e.g., timing, temperature control, dynamics of laser scanner tool) which have to be observed for feasible robot motion control and planning. *Closed-loop control* of the welding process plays a significant role, favoring contactless (primarily optical) means of obtaining feedback [1], due to its low implementation cost, high reliability and short time lag. Low-level robot motion control and higher-level planning have to solve a number of typical problems related to welding applications, from efficient exploitation of physical limits [2], optimized motion transients [5], coordinated motion of robot and scanner with different control properties [6], all the way to a proper path plan minimizing "unproductive" movement or efficient sequencing of welding tasks in the workspace [6]. Algorithms for *task sequencing and robot path planning* are introduced in [15], where task sequencing is performed by solving a traveling salesman problem (TSP) over the seam positions in the Cartesian space. A drawback of the approach is that it ignores detailed geometry, accessibility constraints, and technological parameters. A similar problem, the minimization of processing time in a milling operation, is investigated in [3]. Here, a generalized TSP approach is proposed, where the nodes correspond to the candidate tool entry/exit points for machining a feature.

The nature of laser welding, especially its high processing speed, may also affect long-term decisions regarding the positioning of the workpiece (see [11], [12]) or the layout of the welding cell [7]. Much of these have been subject to intuitive or experience-supported human planning, but recent trends point towards (semi-) automatic support for these once-per-type or once-per-facility decisions, too.

2. Problem statement

In this paper we focus on the *welding cycle* of an assembly cell that includes the workpiece grasped in a fixed fixture and a single RLW robot. Consequently, planning and synchronizing the operations of other equipment that may manipulate the workpiece or monitor the process are out of scope now. We assume that two layers of the same material are welded at a time in a single pass, and the gap of layers is maintained by proper dimpling and fixturing. Welding from two directions is principally possible, but due to the fixture, practically most welds are done from one side only. Hence, a *welding task* is defined as a linear seam together with its *welding direction* which is the surface normal at the middle of the seam. Clamps are placed sufficiently far from the seams to avoid any visibility issues (see Fig. 1). Note that both the welding tasks and the clamping are defined by product design and manufacturing process planning, and are not subject to change in the phase of planning the welding process. In general, we assume that all points of the seams can be accessed within the technologically feasible inclination angle (usually 15º). Hence, *collision check* between the robot and workpiece in the fixture can be managed by post-processing.

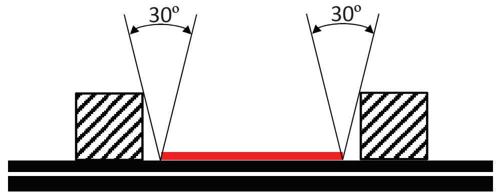

Fig. 1. Visibility of a welding seam

The definition of welding tasks comes together with the *CAD model* of the workpiece. Since the polytope model is sufficient both for determining welding tasks and off-line collision detection via simulation, the 3D geometry is represented in the triangular mesh (STL) format. As for a typical setup, we take the car door, with ca. 50 seams, whose welding tasks are performed by a single RLW robot. In our experiments below we used an industrial smart laser robot (see Fig. 2).

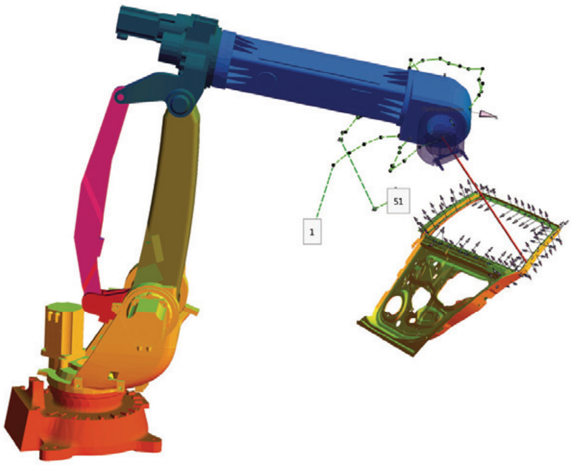

Fig. 2. Workpiece and robot in a typical RLW setting

Summing up, planning takes as *input* (1) the CAD model of the workpiece and its fixture, (2) the specification of the welding tasks, and (3) the geometrical and kinematical model of the robot. The *output* of the planning process is the robot program that controls (1) the joints of the robot (including mirrors in the scanner head) and the (2) laser beam. The final program should comply with the visibility and kinematic constraints, and should also minimize the total cycle time.

3. The planning process

3.1. Workflow of off-line robot programming

The overall robot off-line programming problem has many facets and its solution requires a wide range of expertise from combinatorial optimization up to simulation of robot movements. Hence, it is worth decomposing the problem into multiple phases. We suggest the workflow shown in Fig. 3. Here, the key issue is that task sequencing and path planning happen in the space of the workpiece, still without considering the actual robot. While complete collision check of the robot and its operating environment (including the workpiece) can only be performed after having the robot trajectory, some preliminary check of the possible interaction of the workpiece and the laser beam or the scanner head can be done already during this path planning phase. Nevertheless, we assume that knowledge of actual part geometry can postpone this check to off-line simulation. Next, we discuss the first two planning phases: integrated task sequencing and path planning, as well as inverse kinematic transformation of the robot's path. For our more detailed discussion of the first phase, see also [9].

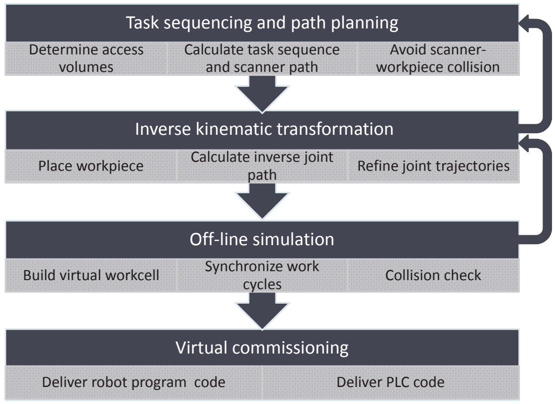

Fig. 3. Workflow of the complete off-line programming process

3.2. Task sequencing and path planning

The problems of welding *task sequencing* and *rough-cut path planning* are strongly related, and must, therefore, be solved together, in an integrated way. Hence, the problem consists of sequencing the individual welding tasks, each corresponding to a single welding seam, and computing a rough-cut robot path, in such a way that the cycle time of the complete welding operation is minimized. It is assumed that there is a set of n welding tasks, denoted as $\{s_1,s_2,\ldots,s_n\}$, to be performed by a single robot in a single operation. Each seam is characterized by its *access volume*, C_i, i.e., a truncated cone defined by the maximal incidence angle and the focus range, and the associated welding time, t_i. We assume that the maximum *robot speed* (speed of the scanner), v, is independent of the position in the working area.

The problem consists of determining the optimal *task sequence*, $(p_1,p_2,\ldots,p_n)$, where $p_i = j$ means that seam s_j is at the ith position in the task sequence, together with the corresponding scanner path. It is easy to observe that the optimal scanner path for a given, fixed sequence is a broken line defined by $2n$ points, $(a_1,b_1,a_2,b_2,\ldots,a_n,b_n)$, where a_i is the position of the scanner when it starts welding seam s_{pi}, the so-called *entry point*, and b_i is the scanner position when it completes welding s_{pi}, the *exit point*. Obviously, a_i and b_i must be inside C_{pi}. Paths with $b_i = a_{i+1}$ are allowed, moreover, this situation reflects an efficient solution in which robot motion and welding overlap completely in that section of the solution. Furthermore, it can be observed that there exists an optimal path where $d(a_i,b_i) \leq t_i v$, and motion in parallel with welding between each pair a_i and b_i takes exactly t_i time. In the sequel, we will restrict our search to such a kind of paths. Motion (without welding) between points b_i and a_{i+1} takes $d(b_i,a_{i+1})/v$ time.

It is assumed that there are no finite acceleration limits, the robot has an infinite working area, and collision checking does not have to be performed at the time of task sequencing, and hence, there are no further constraints that bound the choice of a_i and b_i.

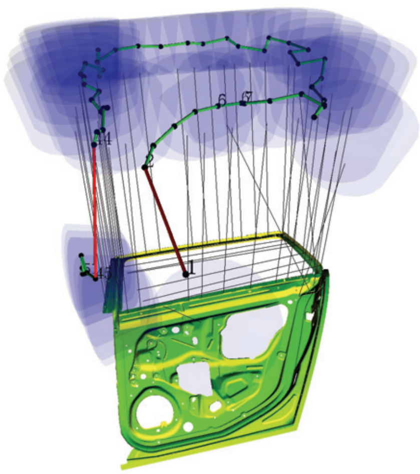

Fig. 4. A solution of the task sequencing problem. The robot moves the laser scanner along the path above the workpiece, and welds the seams from their access volumes, indicated by the truncated cones.

3.2.1. Solution approach

The problem in scope can be considered the direct product of a *traveling salesman problem* (for optimizing the task sequence) and a path planning problem in the 3D space (for finding the corresponding scanner path). For solving this problem, a *tabu search* algorithm has been developed. The algorithm combines adaptations of classical local search operators for TSP for modifying the task sequence, and a path planning heuristics that computes a close-to-optimal scanner path for each candidate task sequence. In each iteration cycle, the next solution is selected according to the rules of the tabu search meta-heuristics. The algorithm terminates when it hits the defined time limit.

3.2.2. Optimizing the task sequence

The initial solution is constructed using an adapted version of the *farthest insertion heuristics* [8]. The algorithm inserts the seams one-by-one into the sequence: in each iteration cycle, it considers the seam that is the farthest from the current path, and inserts it into its locally best position.

During the improvement phase, the tabu search selects the best non-tabu solution from the neighborhood of the current solution. We applied the so-called *2-opt* (deleting two edges and re-connecting the tour) and *or-opt* (relocating a segment of the tour of maximal length k to another position, in forward or backward orientation) neighborhoods [8]. Several *filtering* techniques have been implemented to eliminate members of these neighborhoods that cannot improve the solution. The evaluation of the neighbor involves computing a new scanner path for the modified task sequence. When a move is made, the edges deleted from the tour are added to the tabu list, and a subsequent move is declared tabu if it would reinsert such an edge, unless the resulting solution is better than any previously encountered solution. For a detailed presentation of the above solution techniques for TSP, the reader is referred to [8].

3.2.3. Computing the scanner path

The path planner algorithm computes a close-to-optimal scanner path for each task sequence evaluated during the tabu search. An incremental algorithm is applied that departs from the path computed for the original solution, and adapts it to the changes performed by the neighborhood function. The algorithm sweeps along the broken line for a fixed number of iterations, and adjusts a single corner point of the broken line at a time. The new position of the entry point a_i (exit point b_i) is determined in such a way that it is inside the corresponding access volume, sufficiently close to the exit point b_i (entry point a_i), and minimizes the distance $d(a_i, b_{i-1})$ (resp., $d(b_i, a_{i+1})$).

3.3. Inverse kinematics

Having once obtained the path of the laser scanner in Cartesian space of the workpiece, the corresponding robot motion—i.e., a motion sequence prescribed for the joint variables—has to be calculated. First, the workpiece is placed in the workspace of the robot. Now this *placement* problem is solved by engineering heuristics, i.e., by putting the workpiece into the centre of the workspace of the robot, ensuring thus that the workspace includes the access volumes of all tasks. The general *inverse kinematics* problem would involve a 6-DOF manipulator with a 6-dimensional, i.e., fully specified task description. However, in the RLW case we have, in fact, seven kinematic DOF: four “conventional” revolute joints, two tilting mirrors (counting as revolute joints), and one prismatic DOF in the form of the focal length of the welding beam. Considering the dimensions alone, this would result in an underspecified task, requiring *redundancy resolution*.

However, specific features of the RLW problem allow a convenient workaround. Our application case specifies both the scanner position and the location of the corresponding welding point on the workpiece, the latter being considered the *tool center point* (TCP) of the robot. This form of task definition does already determine the focal length of the beam, thereby assigning a specific value to one DOF of the robot. Now, we have to handle one more redundant DOF—this is done by setting the angle of the last tilting mirror to a mid-range value. The justification for the latter is the extremely narrow free range of the last joint—it is best to leave it mid-range to prepare for any additional angle compensation during execution. Having thus resolved kinematic redundancy, we may now proceed with an inverse kinematic solution for each prescribed point of the rough-cut path.

Solving the inverse kinematic problem implies the transformation of task space position to adequate *joint values*. Closed-form solutions (obtained in symbolic form with robot-specific parameters substituted during use) are to be preferred, as far as kinematic properties allow, due to their accuracy and low computational demand. Obtaining a closed-form solution for a part of our manipulator relies on *template equations* (see, e.g., Paul *et al.* [14]). This approach exploits the fact that the solution of the inverse kinematic equations of most industrial manipulators boils down to solving trigonometric polynomials following simple patterns, also referred to as *template* or *prototype equations*. If their symbolic solutions are known beforehand, finding a closed-form solution for an entire manipulator will consist in 1) identifying the matching templates, and 2) using them to generate the inverse solution in a purely symbolic way. We apply the *Linkage Designer* package of *Mathematica* that supports template-based solution in a semi-automatic way: possible matches are found automatically, while their verification requires human intervention [13]. This is a reasonable compromise, since a closed-form solution has to be generated only once for a given manipulator class, and can be used in several applications after substituting the parameters specific to a given robot.

The properties of our welding robot do, however, not allow a closed-form solution for the entire kinematic chain, due to the last revolute joint (in our case, the laser mirror) having an offset in an unfavorable direction, as shown in Fig. 5. Even in this case, much of the inverse kinematic problem can be analytically solved, narrowing down the space subject to *iterative search* to only *one dimension*.

In order to obtain a partial closed-form solution, we first solve the inverse kinematic problem for a fictitious robot which does not have an offset in the last revolute joint. For that we take a beam exiting the robot *without offset* (marked green in Fig. 5). It is known that the offset is always perpendicular to the fictitious beam, therefore, the latter must always be this fixed offset away from the beam actually emitted (marked red in Fig. 5). Hence, the TCP would always lie somewhere on the rim of a circle of known radius around the fictitious beam (circle marked light green in Fig. 5). Tracing this circle with the zero-offset beam under unchanged orientation would, at one point, bring the actual beam (and thus, the actual TCP) into the originally desired position. This can be conveniently found by *one-dimensional search*. Two solution branches exist for our robot (one "left-handed" and one "right-handed" configuration), implying the need for two searches; however, preceding solutions would usually make a sound suggestion for a given configuration branch.

Fig. 5. Closed-form solution for a kinematic chain without offset (green beam), and one-dimensional iterative search for the real TCP (red beam).

The solution method is thus summarized as follows:

1. Obtain specification of 1) the scanner points along its path, together with 2) the related positions of the welding seams on the workpiece. The required beam length is, hereby, already fixed; now we can solve for known scanner position and zero beam length, and apply actual beam length separately.
2. Solve the inverse kinematic problem for fixed beam length and mid-range angle of last mirror.
    a. Generate *closed-form solution* for fictitious robot with zero mirror offset and zero beam length.
    b. Find left-handed or right-handed solution for actual mirror offset and prescribed beam length via *one-dimensional iterative search*.

4. Computational experiments

The proposed task sequencing and path planning algorithm has been evaluated in computational experiments on problems involving the assembly of car doors using RLW. Four door designs have been considered, each involving ca. 50 welding seams. All process parameters were set to match a realistic industrial setting. Three algorithms have been compared: the proposed algorithm, which performs integrated task sequencing and path planning, denoted as TS–PP; the algorithm of [15], which solves a TSP over the seam positions and computes the robot path afterwards (RMV); and a modified version of RMV that solves the TSP over the midpoints of the access volumes, instead of the seam positions (RMV*). The algorithms have been implemented in C++, and the latter two algorithms used IBM ILOG CP as a TSP solver. A time limit of 60 seconds was applied.

Table 1. Computational results.

	TS–PP		RMV		RMV*	
	cycle	idle	cycle	idle	cycle	idle
Part1	23.65	3.25	51.86	31.46	24.87	4.47
Part2	27.6	6.8	94.66	73.86	30.34	9.54
Part3	30.46	9.66	54.31	33.51	32.74	11.94
Part4	29.23	8.43	149.35	128.55	32.27	11.47
Avg.	27.735	7.035	87.545	66.845	30.055	9.355

The results are summarized in Table 1 which displays the overall cycle time and the idle time (part of the cycle time when the laser beam is switched off) in seconds for each algorithm and each workpiece. The results show that TS–PP outperformed the other algorithms on all instances. In particular, it became obvious that a task sequence computed based solely on the seam positions is unsuitable for workpieces with complex geometry, since it leads to the scanner head moving in a zigzag above seams that have nearby positions but different surface normals. Consequently, RMV resulted in up to 5 times higher cycle times and up to 15 times higher idle times than TS–PP. RMV* performed significantly better than RMV, but still achieved 5–10% higher cycle time and 24–40% higher idle time than the proposed TS–PP algorithm. This gain can be regarded as the benefit of integrating task sequencing and path planning. The time of calculating the inverse path for the sample door with 51 seams was 2.3 seconds. Detailed experimental evaluation of the inverse kinematics is still underway.

5. Conclusions and future work

The paper presented a novel approach for generating off-line programs for RLW robots with the objective of minimizing cycle time. The crux of the planning problem was solved for a conjoint task sequencing and path planning problem defined over the access volumes of welding tasks. Further on, the inverse kinematics was solved by exploiting the features of RLW technology. Currently, we investigate the subproblems of collision checking and placement. Later on, extension to body-in-white will also be considered. This problem involves not only more complex geometries and collision tests, but also the cooperation of multiple welding robots.

Acknowledgement

This work has been supported by EU FP7 grant RLW Navigator No. 285051 [16], and the NKTH grant OMFB-01638/2009.

References

[1] Abt F, Heider A,. Weber R, Graf T, Blug A, Carl D, Höfler H, Nicolosi L, Tetzlaff R. Camera based closed loop control for partial penetration welding of overlap joints. *Physics Procedia* 2011; **12**(A):730–738.

[2] Brogardh T. Present and future robot control development—An industrial perspective.*Annual Reviews in Control* 2007; **31**:69-79.

[3] Castelino K, D'Souza R, Wright PK. Toolpath optimization for minimizing airtime during machining. *Journal of Manufacturing Systems* 2002; **22**(3):173–180.

[4] Ceglarek D, Huang W, Zhou S, Ding Y, Kumar R, Zhou Y. Time-based competition in manufacturing: stream-of-variation analysis (SOVA) methodology – review. *International Journal of Flexible Manufacturing Systems* 2004; **16**(1):11-44.

[5] Gasparetto A, Zanotto V. Optimal trajectory planning for industrial robots. *Advances in Engineering Software* 2010; **41**:548–556

[6] Hatwig J, Reinhart G, Zaeh MF. Automated task planning for industrial robots and laser scanners for remote laser beam welding and cutting. *Production Engineering* 2010; **4**(4):327–332.

[7] Iordachescu D, Blasco M, Lopez R, Cuesta A, Iordachescu M, Ocaña JL. Development of robotized laser welding applications for joining thin sheets. *Proc. of 2011 International Conference on Optimization of the Robots and Manipulators*, 1–5.

[8] Johnson DS, McGeoch LA. The traveling salesman problem: A case study in local optimization. In Aart EHL, Lenstra JK (eds), *Local Search in Combinatorial Optimization*. 1997, John Wiley and Sons, Ltd. 215–310.

[9] Kovács A. Task sequencing for remote laser welding in the automotive industry. *23rd International Conference on Automated Planning and Scheduling, ICAPS'13*, p. 6, accepted.

[10] Maropoulos PG, Ceglarek D. Design verification and validation in product lifecycle. *CIRP Annals – Manufacturing Technology* 2010; **59**(2):740-759.

[11] Michalos G, Makris S, Papakostas N, Mourtzis D, Chryssolouris G. Automotive assembly technologies review: Challenges and outlook for a flexible and adaptive approach. *CIRP Journal of Manufacturing Science and Technology* 2010; **2**:81–91

[12] Mitsi S, Bouzakis KD, Sagris D, Mansour G. Determination of optimum robot base location considering discrete end-effector positions by means of hybrid genetic algorithm, *Robotics and Computer-Integrated Manufacturing* 2008; **24**(1): 50–59.

[13] Müller M, Erdős G, Xirouchakis P. High accuracy spline interpolation for 5-axis machining. *Computer Aided Design* 2004; **36**:1379–1393.

[14] Paul RP, Shimano B, Mayer G. Kinematic control equations for simple manipulators. *IEEE Trans. on Systems Man and Cybernetics*, **11**(6):449–455, 1981.

[15] Reinhart G, Munzert U, Vogl W. A programming system for robot-based remote-laser-welding with conventional optics. *CIRP Annals – Manufacturing Technology* 2008; **57**(1):37–40.

[16] Remote Laser Welding (RLW) System Navigator for Eco and Resilient Automotive Factories, FoF-ICT-2011.7, No. 285051, http://www.RLWnavigator.eu/, accessed on 05.03.2013.

[17] Shibata, K. Recent automotive applications of laser processing in Japan. *The Review of Laser Engineering* 2008; **36**:1188–1191.

[18] Tsoukantas G, Salonitis K, Stournaras A, Stavropoulos P, Chryssolouris G. On optical design limitations of generalized two-mirror remote beam delivery laser systems: The case of remote welding. *International Journal of Advanced Manufacturing Technology* 2007; **32**(9–10):932–941.